

Classic Time-stamped Data

Correlation assistance

Mon, Mar 12, 2001

When using the Classic protocol to acquire data from IRM front-ends, one can easily keep up with 15 Hz, because the client software is aware of the arrival of the time of arrival of new reply data. But if the request for data involves replies from many nodes, it can be more difficult to determine when one has correlated data available across front-ends. A time-stamp ability can sort this out by labeling a reply to a 15 Hz cycle. This note explores how to do this.

Time-stamp vehicle

For the Fermilab 15 Hz accelerator components, data acquisition is designed around the 15 Hz acceleration cycle. The fundamental key to assign one set of data is the 15 Hz cycle number. To use time-of-day as a reference cannot work, unless one also maintains matching time-of-day values for each 15 Hz cycle. This is because the accelerator is not synchronized to time-of-day, but rather to the 60 Hz power line frequency, which varies during the day to suit the needs of the nation's electrical grid.

Time-stamp analog channel

One scheme is to include with each data request an analog channel whose reading value is a two-byte cycle number that can serve to identify a specific 15 Hz cycle, at least within a half hour or so. If the data sought is not analog reading data, one cannot include a request for this cycle number channel in the same data request; however, one can add a separate request (also using the same request period specification) that is included in the same datagram with the data request of interest. In that case, the client support software must be made aware of the arrival of the entire datagram so it can associate the cycle number value with any other data request whose reply is included within the same datagram. If one used this scheme, it might be attractive to consider dedicating channel 0000 for this purpose.

Time-stamp listype

If only a few types of data needed time-stamp information, one could define one or more listypes for this purpose. The format of the reply data would include a time-stamp cycle number along with the data of interest. Take one example, that of analog readings. If one made a data request for readings of analog channels to be updated at 15 Hz, the reply data could be the reading data word following by the cycle number. (It could also be the other way around.) One would need to make a data request asking for 4 bytes of data rather than 2, of course. If many channel readings are sought, the size of the reply data would therefore be twice as much, with all readings associated with the same cycle number value.

Reply status word

In the Classic protocol reply message is included a status word. At this time, only nonzero values from 1–8 are used. One could consider including time-stamp values in this word. If we used the low byte of the word for the actual status, the high byte could contain the low byte of the cycle number. Although these values repeat every 17 seconds, it should be enough to easily sort out replies from multiple front-ends. A disadvantage of this approach is that client software may have to be modified that now considers any nonzero status word to be a problem. In this case, one would expect to get reply values of cc00, where cc is the current 8-bit cycle number. An advantage of this approach is that it can mix with any type of data and thus with any data request.